HWK

Preamble

hwk, named after the raptorial bird, is a wireless LAN penetration and stress-testing tool. It provides various modes such as the new experimental
probe response fuzzing, beacon injection as well as simple authentication and deauthentication flooding to name a few.
Furthermore, it allows you to simply have a look at what's happening in the air or just to align your antenna for best signal strength.
hwk also works against WPA/WPA2 encrypted networks because it does not break the encryption, hwk just exploits the protocol weaknesses of the MAC layer for example unverified deauthentication packets. It also does automatically enable the monitor mode.

Requirements

hwk uses the libpcap library to send and receive network packets, so it would be great if you install libpcap before compiling hwk :)
Of course you need a wireless interface which is capable of the monitoring mode.
Working injection is also not bad, but hwk will do some tests!

- libpcap
- wireless device chipset which supports monitor mode and injection

Setup

1. Download the sources from hwk.sf.net.
2. cd ~/Downloads/hwk
3. unpack it using for example tar -xvzf
4. take a look on include/hwk.h (currently line 45) and add/remove
   some channels if they're not supported by your card.
   To check which channels are supported simply run a 'iwlist <iface> channel'

3. run 'make'
4. run 'make install' (as root)
5. run 'make clean'
6. That's it!

Note: use "make uninstall" to remove hwk from your system.

Modes

hwk supports various modes. This chapter will only give a short summary about them, details are described below.
One of these options has to be set, otherwise hwk defaults to --scan.

```
--help:        show a small help
--scan:        scanning mode, shows the captured packets and
               prints out mac addresses and other details
--deauth:      single deauthentication flood
--auth:        authentication flood
--focus:       antenna alignment
--fuzzprobes:  probe response fuzzing
--testinject:  injection testing
--beacon:      beacon flood
--9:           multi deauthentication flood
```

Scanning mode (--scan)

Scanning mode is used to gather information of active APs in the area. Gather information means in this case:

> [+] get the relevant MAC addresses
> [+] reveal the data connections (and get their MACs)
> [+] see who is sending probe requests
> [+] determine the channel an AP is sending on

It shows also the encryption of a network you may identify via 'CRYPT' set on a data packet. Other flags are 'FromDS' and 'ToDS' which means 'packet is sent by the AP and goes out to the receiver'. 'ToDS' vice versa.

You may also 'grep' the output of hwk:
```
$ hwk -m scan | grep DATA
```

This for example would only show the data packets. It's sometimes useful especially if you exactly know your target AP and you just want to have stations which are connected to that AP.

Single deauthentication flood (--deauth)

Single deauth flood constantly sends deauthentication packets with the source mac address of one specific client.

Required options: `{ --bssid, --dest, --client }`
Optional: `{ --channel }`

If no channel is given, hwk does automatically try to detect the channel. Unfortunately, this fails sometimes xD

When launching this attack, hwk performs injection tests (they might fail if mac address filtering is enabled) and sequence number catching. If injection fails, either your card is unable to support injection or is too far away from the target AP.
The given [INJ] flag indicates possibly failed injection.

It's always useful to have a look on the ACKed (acknowledged) packets as they indicate if your attack successfully runs. If the number of sent packets is similar to (or greater than) the ACKs it means that your deauthentication packets successfully received.

Authentication flood (--auth)

Launch an authentication flood against a specific AP.

Required options: `{ --bssid, --dest, --client }`
Optional: `{ --channel }`

In fact, this mode is very similar to the deauthentication flood but it sends spoofed authentication packets: the classical DoS.
Another point which is important to mention here is that this mode gains more effect when combined with --extreme (extreme flood) to increase the packet flow. Authentication flood doesn't require an associated client on the AP which is the target to be deauthenticated.

Focus mode (--focus)

Focus mode is useful for antenna alignment by showing the signal strength of a specific sending station. You may try out different antenna positions to determine which one is the best!

Required options: `{ --channel, --client }`

Injection Testing (--testinject)

Injections testing helps you to determine if frame injection works correctly on the given channel or (default) all channels.

Optional arguments: `{ --channel, --bssid }`

Probe Response Fuzzing Mode

Probe Response Fuzzing injects different packets which have been filled up using random tags (such as channel, country code etc) aiming at driver implementation weaknesses.

Required options: `{ --channel, --bssid }`
Optional: `{ --dest }`

Beacon fuzzing mode (--beacon)   [updated]

Beacon fuzzing aims at the same potential security flaws as Probe Response fuzzing: insecure written drivers. As mentioned at Deepsec, you may use this mode in a virtual environment to increase efficiency!

Required options: `{ --channel, --bssid }`
Optional: `{ --dest }`

Multi Deauthentication (--9)

Multi Deauthentication listens for `--delay=seconds` (default: 25) using channel hopping, detects various data connections (Data + QoS Data) and afterwards sends deauthentication packets to all of the detected connection peers.

Optional arguments: `{ --channel }`

Other arguments explained

```
--channel=<channel>      specify channel,
                         deactivate channel hopping
--extreme                extreme flood mode
                         sends as much packets as possible
                         (use it with auth flood xD)
--iface=<iface>          set interface (e.g. wlan0/ath0)
                         monitor mode will be enabled automatically
--delay=<time>           ONLY for -9-mode; specify the time to
                         gather data connections
```

Flags

Flags which are used to indicate possible problems or warnings are used by various modes, they're often showed behind the status line.

```
E:          extreme flooding is enabled
            (this may influence other WLANs as an unwanted result)
INJ:        Injection test failed. This might have different possible
            reasons:
            - AP too far away
            - card does not support injection
```

EOF